Cyber Dumpster Diving – creating new software systems for less

Ian Gorton, Lab Fellow,
Group Lead, Data Intensive Scientific Computing,
Computational Sciences and Math Division
Pacific Northwest National Lab

PNNL

- Department of Energy Science Lab
  - Physical and fundamental sciences
  - National security
- 4500+ people
- Business volume of over $1b per annum
- Large scale experimental facilities, e.g.
  - Environmental Molecular Sciences Lab (EMSL)
  - 161 Tflop supercomputer

Pacific Northwest
NATIONAL LABORATORY

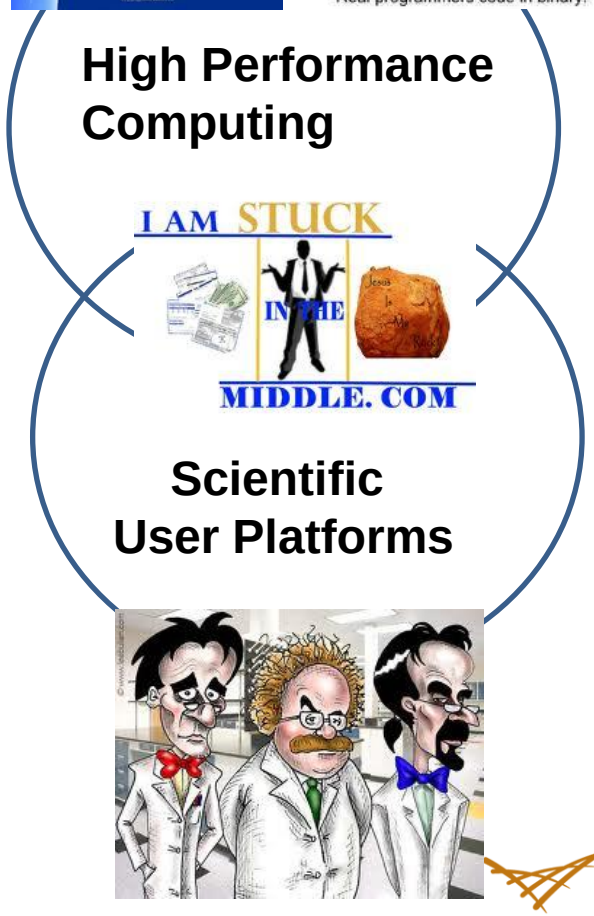
The most radical possible solution for constructing software is not to construct it at all.

Fred Brooks: No Silver Bullet: Essence and Accidents of Software Engineering

Pacific Northwest
NATIONAL LABORATORY

DISC@PNNL

- Data Intensive Scientific Computing
  - User platforms
  - Data management
  - Tool integration
  - Workflows
  - Provenance
- Applications in e.g.
  - Bioinformatics
  - Climate modeling
  - Carbon sequestration
  - Subsurface modeling

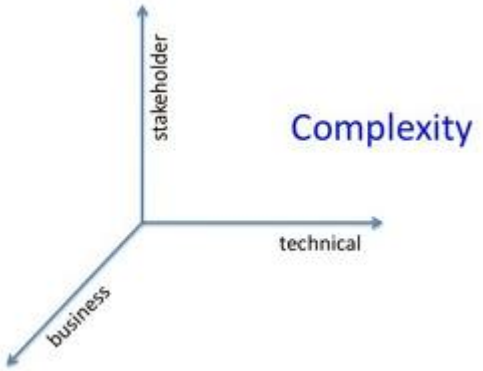


Pacific Northwest
NATIONAL LABORATORY

The middle is a hard place …

- Requirements
  - Need to understand science domain
  - Need to understand HPC
  - Difficult to define, constant refinement, negotiate, negotiate
  - “The hardest single part of building a software system is deciding precisely what to build.”
- Design
  - Conflicting quality requirements
  - Complex, heterogeneous technologies
  - Large data
  - Proliferation of tools, variable quality

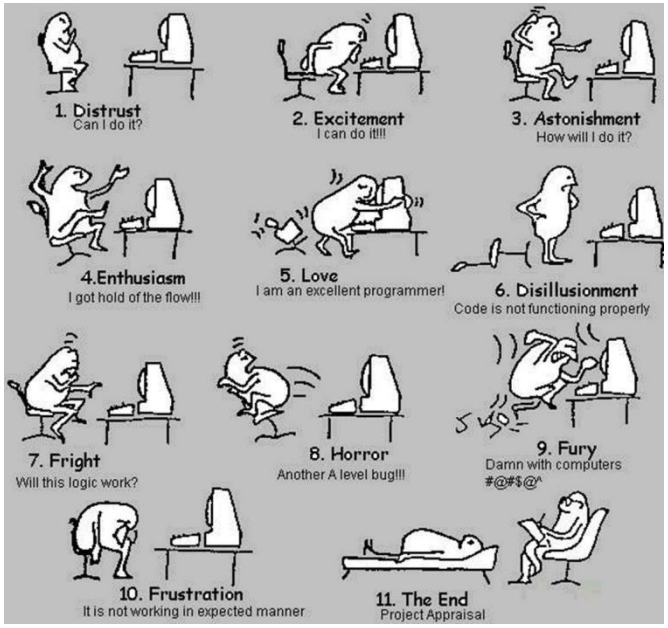
“Would've been cheaper to pay the ransom.”



Pacific Northwest
NATIONAL LABORATORY

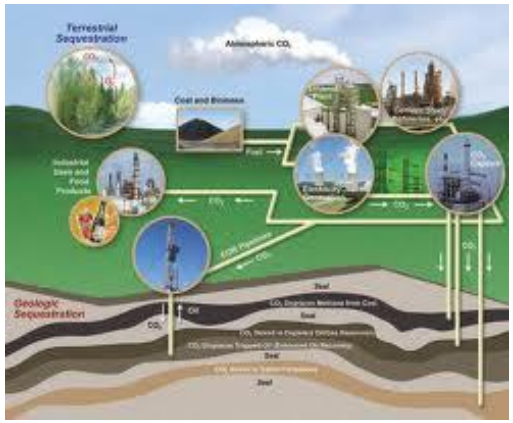
Project Funding Profiles

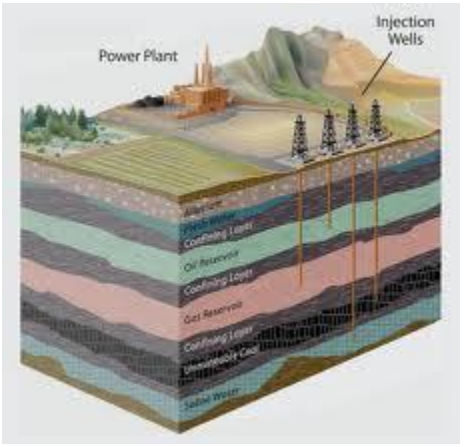
- Typically fixed amounts
  - What can we build with X dollars?
  - Fixed amounts per year, 1-3 year lifecycle
- Limited funding
  - From .25 to 10 team size per year
  - 1-2 people per year most common
- High expectations
  - Scientists think ‘software is easy’
  - it’s just coding, right?

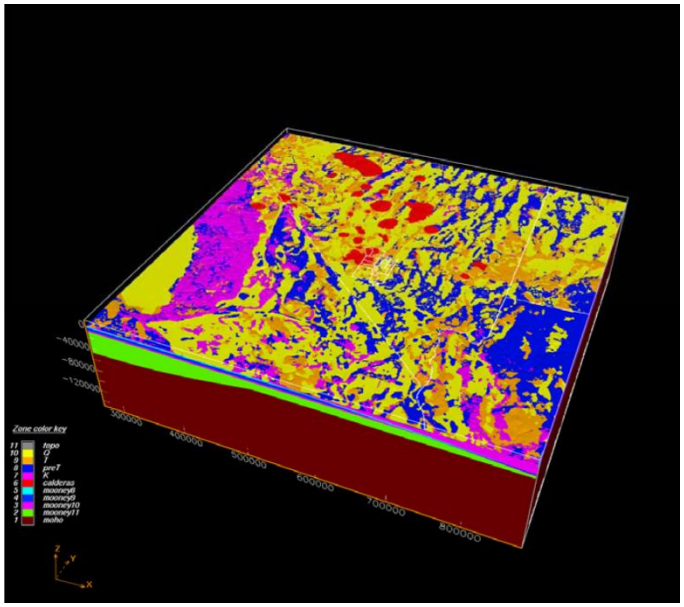


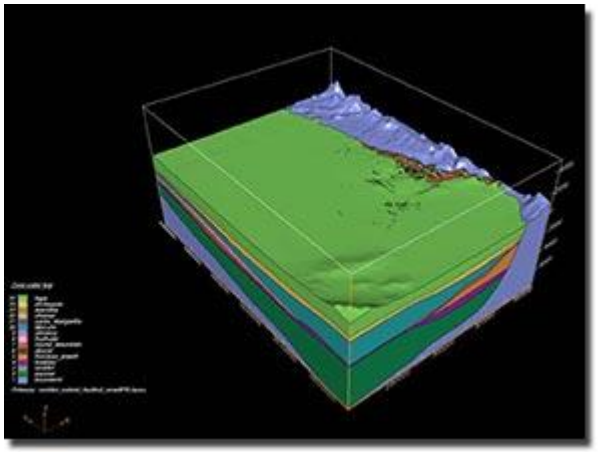
Some Examples

Carbon Sequestration (Storage)



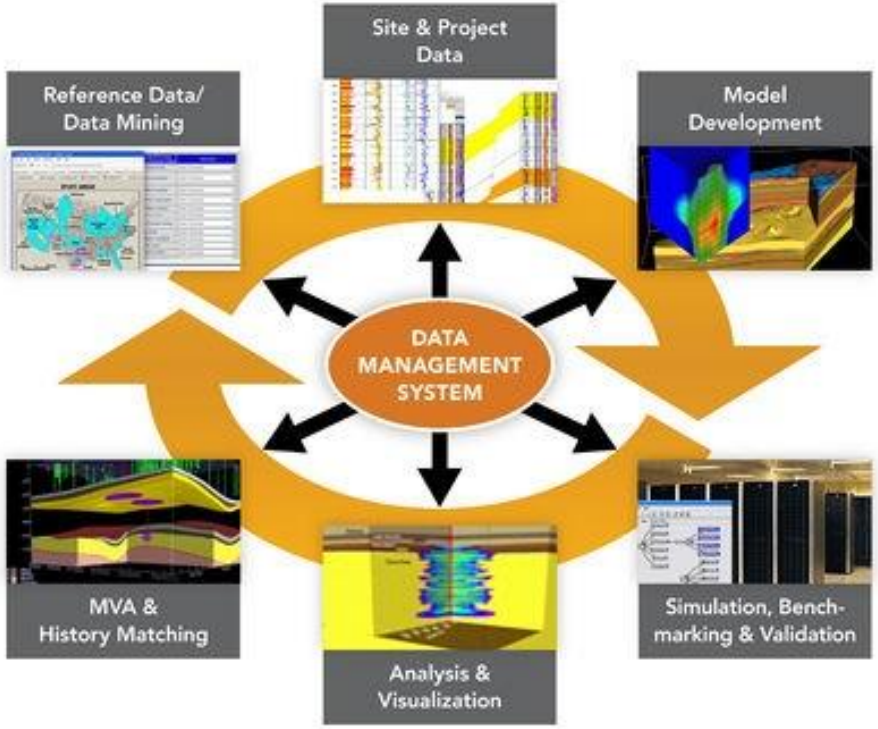






Pacific Northwest
NATIONAL LABORATORY

Geological Sequestration Software Suite (GS3)

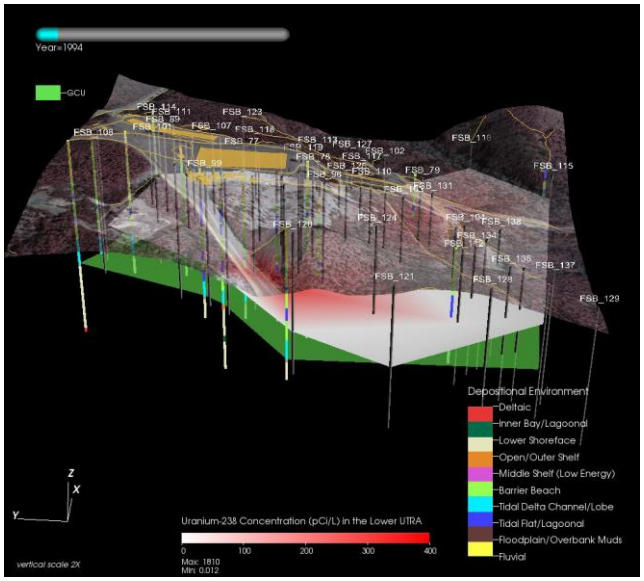


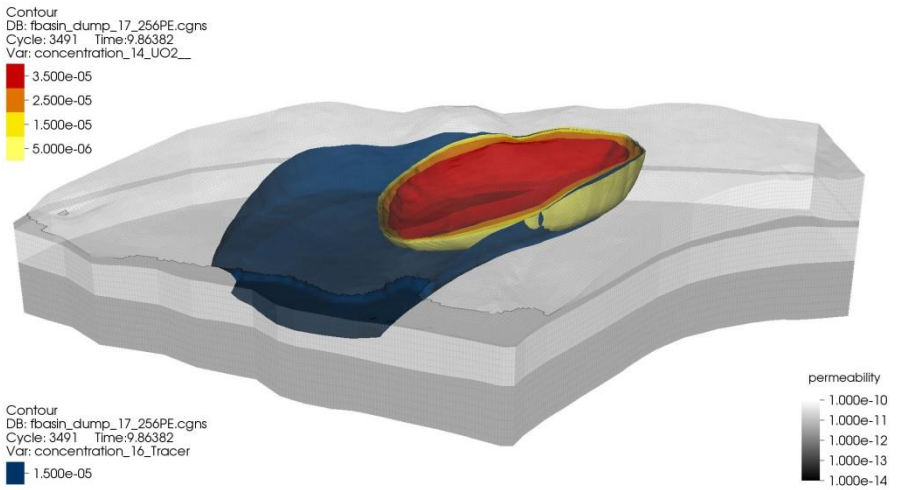
- Large-scale, complex data
  - Experimental
  - HPC Simulation inputs/outputs
  - Multiple realizations for UQ
- Long-lived projects
  - Modeling
  - Analysis
  - Monitoring (100+ years)

Pacific Northwest NATIONAL LABORATORY

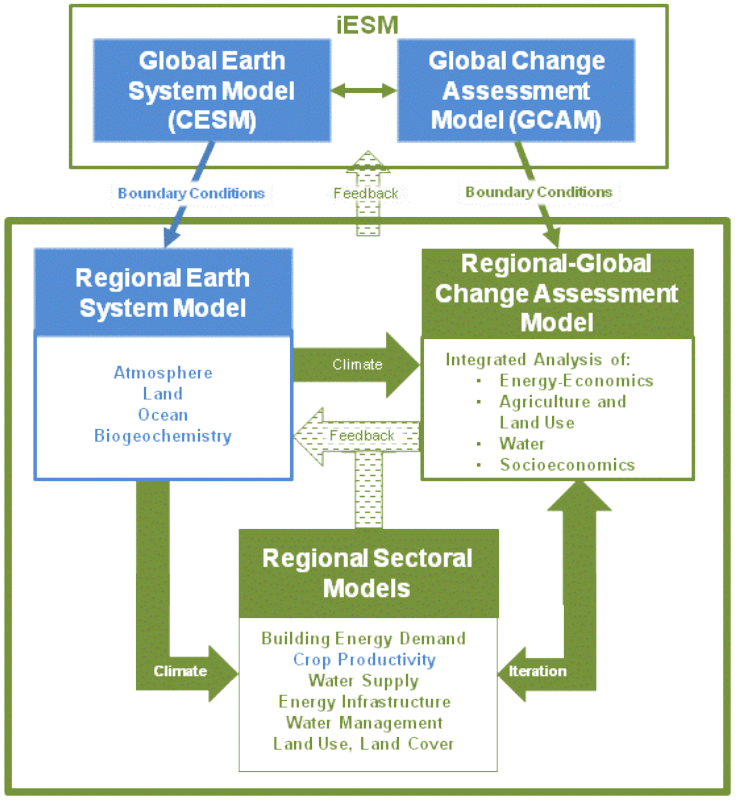
Advanced Simulation Capability for Environmental Management (ASCEM)

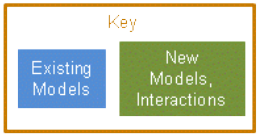
- A State-of-the-art tool for predicting contaminant fate and transport through natural and engineered systems
  - Open source
  - Modular and extensible
  - ‘born’ parallel for execution on emerging architectures.
- A User Platform to manage data, create models and analyze results





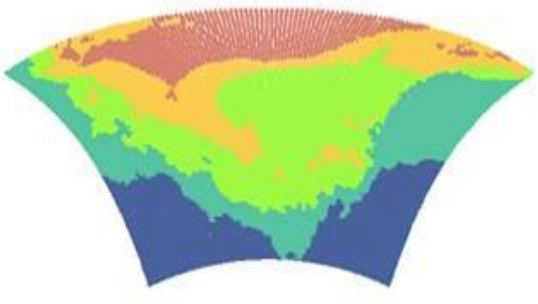
Integrated Regional Earth System Model





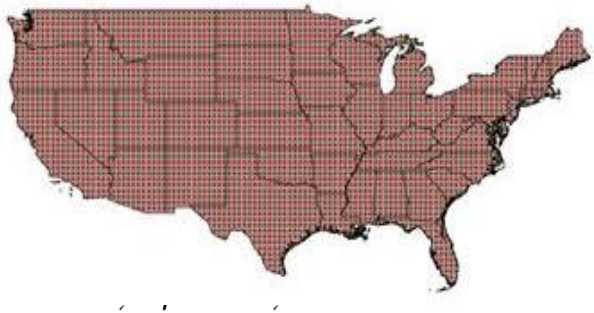
- **Key Attributes:**
  - Modularity
  - Portability
  - Open-Source
  - Integrated Regional Analyses

5



+

=



A powerful, usually legal, source of information that isn't seriously defended because of social taboos.

‘Write-as-little-as-possible’ Reuse

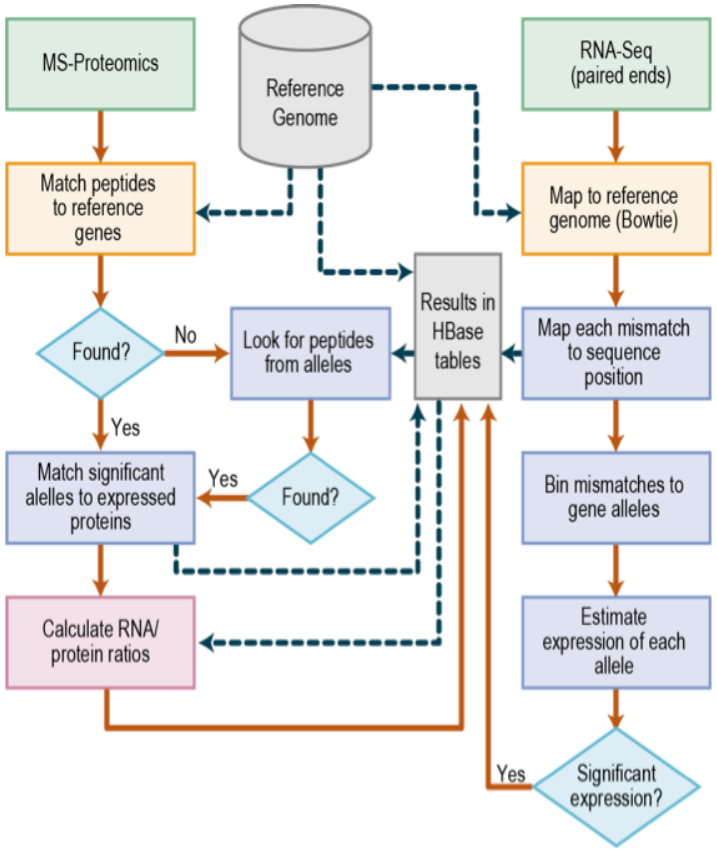
- Approach:
  - Leverage open source frameworks and tools
  - Extend to support science applications
  - Generalize to support multiple science domains
- Requires:
  - Careful technology selection
  - Creative design
  - Robust architectures

Pacific Northwest
NATIONAL LABORATORY

An simple example - bioinformatics

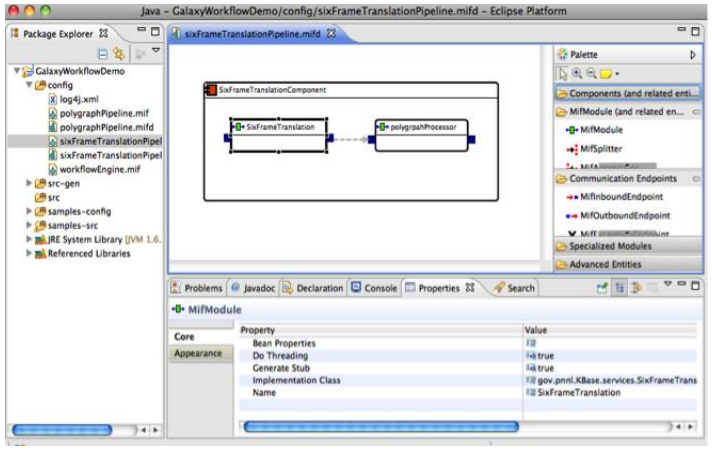
Bioinformatics Workflows

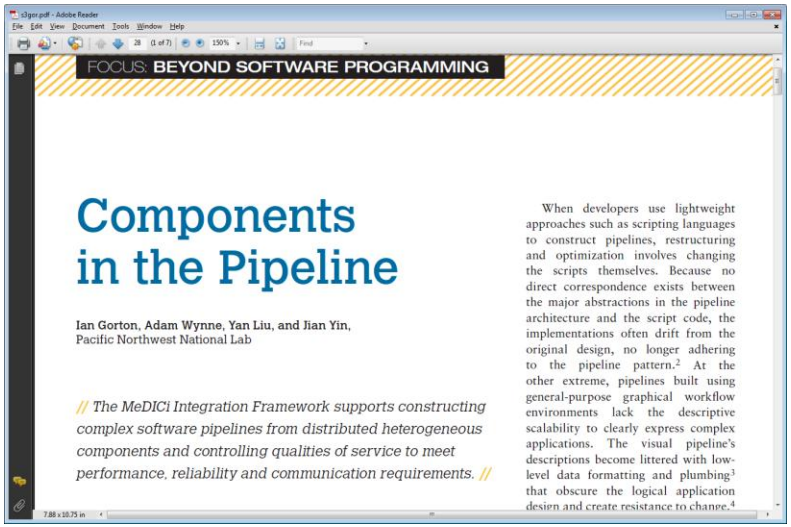
- Biologists want to create computational workflows:
  - Integrate disparate tools and data
  - Span multiple execution platforms
  - Execute reliably and efficiently
  - Capture provenance
- Workflow tools for biology
  - Galaxy
  - Taverna
- Limitations in handling large scale data and computations

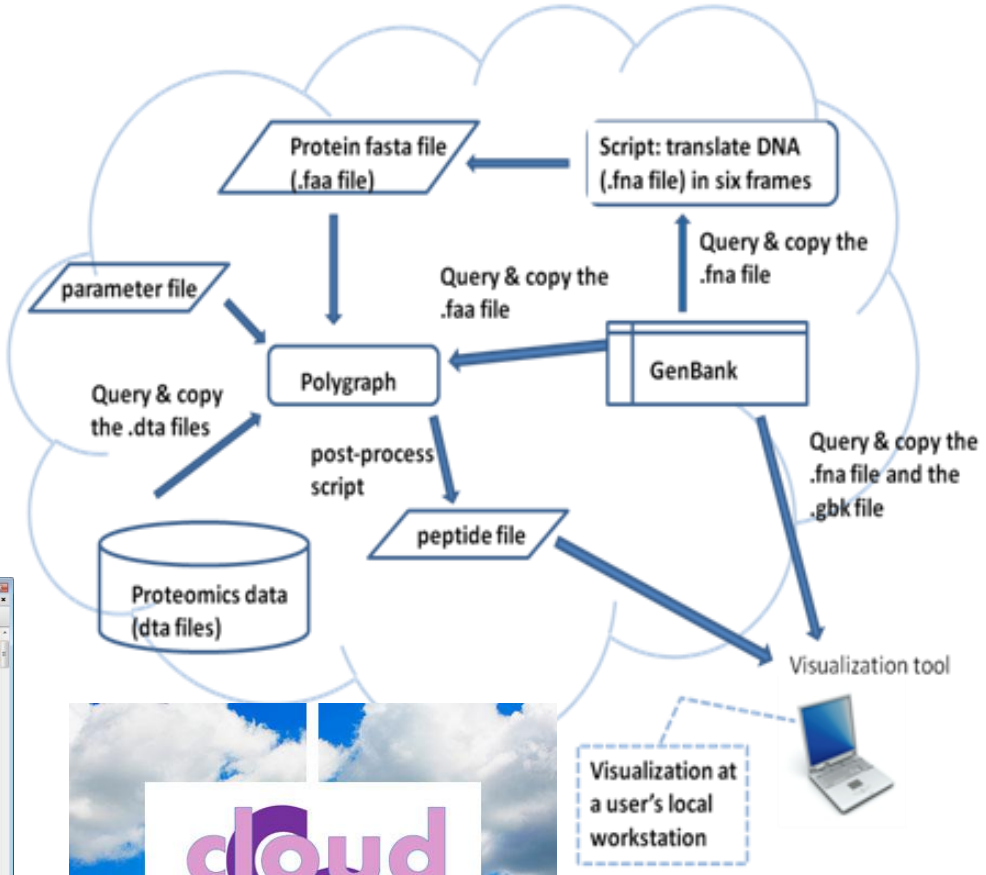


Pacific Northwest NATIONAL LABORATORY

Computational Pipeline Infrastructure







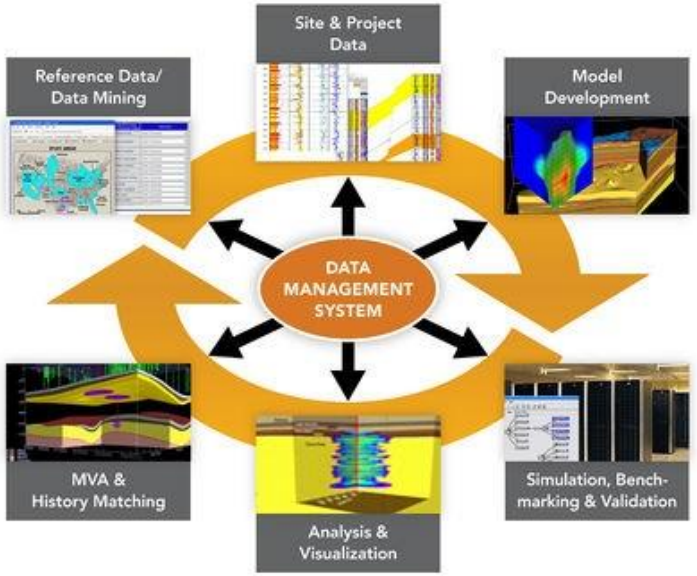
http://www.computer.org/portal/web/csdl/doi/10.1109/MS.2011.23

Pacific Northwest NATIONAL LABORATORY

Velo – Knowledge Management for Modeling and Simulation

Supporting Carbon Sequestration Modeling

- Requirements
  - Collaboration
  - Sharing data
  - Metadata management
  - User-driven customization
  - Extensibility
  - Model versioning
  - Provenance
  - Robust, scalable
- Small project, team ~1.75 people, 3 years



Cyber Dumpster Diving

- Open source
- Candidate technology assessments:
  - Quality of docs
  - Release schedule
  - Community scope
  - APIs
  - Code/architecture
  - Install and workout, simple tests

Pacific Northwest NATIONAL LABORATORY

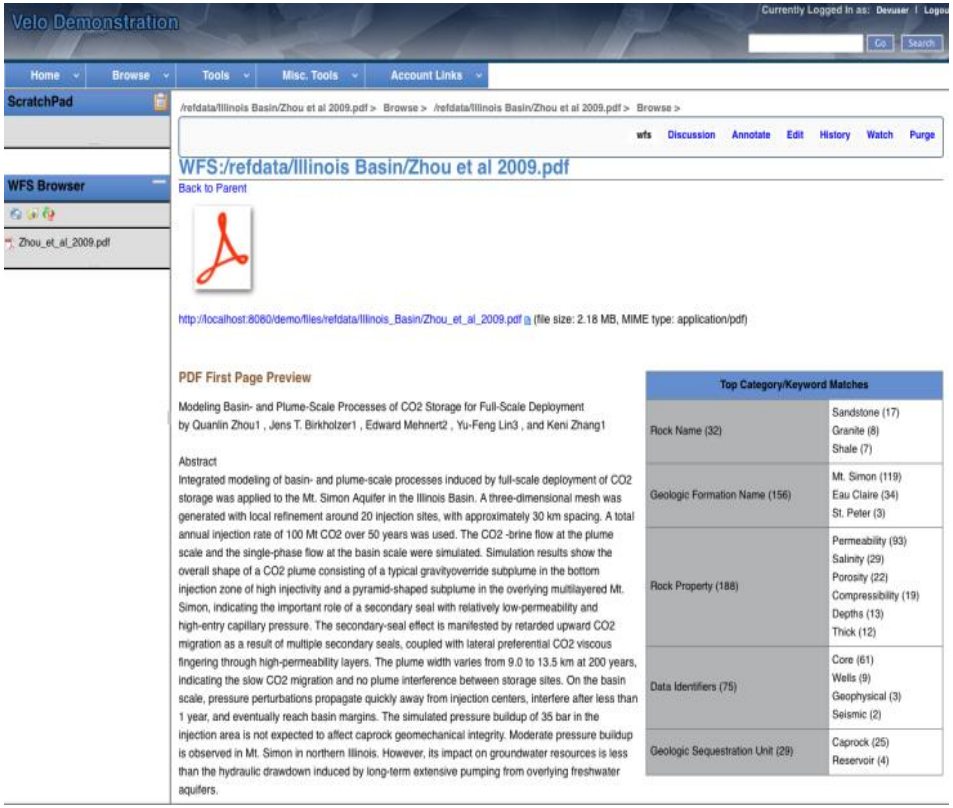
Proudly Operated by Battelle *Since 1965*

Feature-Reuse Matrix

Feature	Solution	Notes	Reuse
Collaboration	Mediawiki	Core wiki features support this	100%
Sharing data	Mediawiki Alfresco	Requires integration of MW and Alfresco	60%
Metadata management	Mediawiki Alfresco	Requires customization of MW and Alfresco basic features	80%
User-driven customization	Mediawiki	Core wiki features support this	100%
Extensibility	Mediawiki Alfresco	APIs support extension, but requires design of exact integration mechanisms	20%
Model versioning	Mediawiki Alfresco	Minor extensions for MW/Alfresco capabilities	75%
Provenance	Mediawiki	Some for free in MW, but advanced features need developing	20%
Role-based Security	Halo ACL	Mediawiki extension	100%

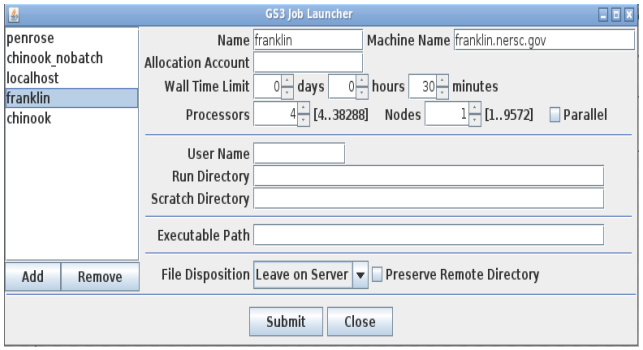
GS3 Examples - Semantic Capabilities - Metadata Extraction

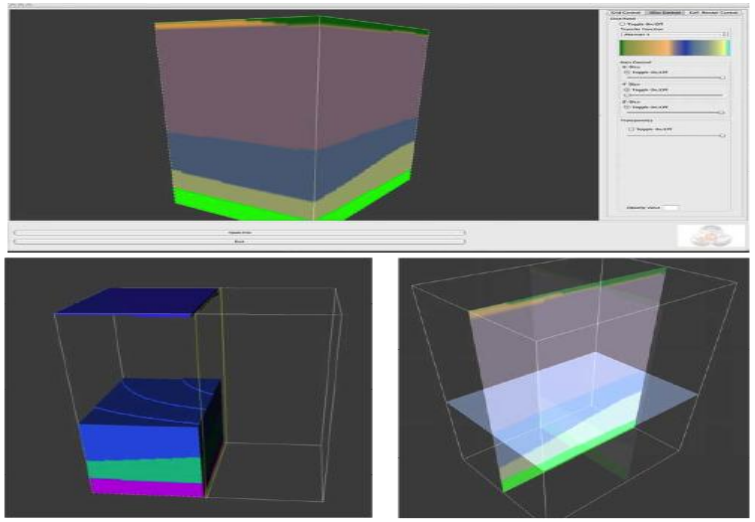
- Metadata:
  - Generic information e.g.file size, owner, preview/thumbnails
  - Specific to the file type, e.g. keywords, geographic location
- Metadata is searchable
- Extensible architecture for custom data types ingest pipelines, e.g.
  - Simulation outputs
  - Spreadsheets
  - Input files



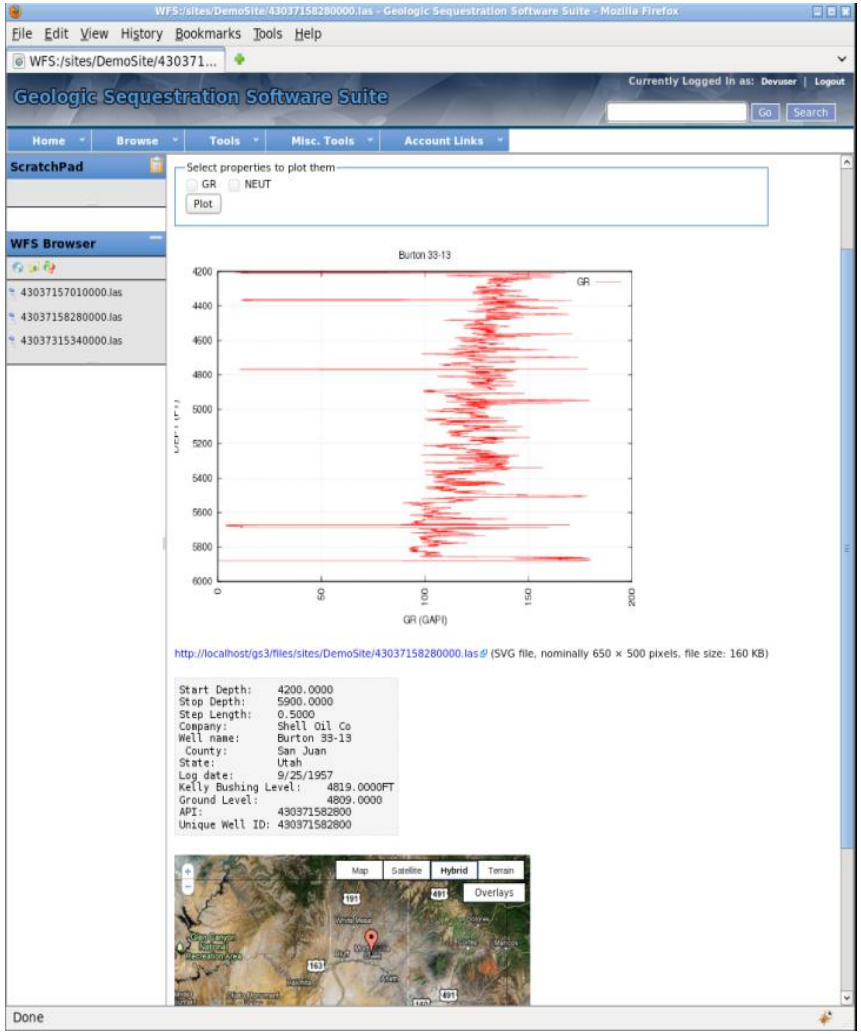
GS3 Examples - Tool Integration

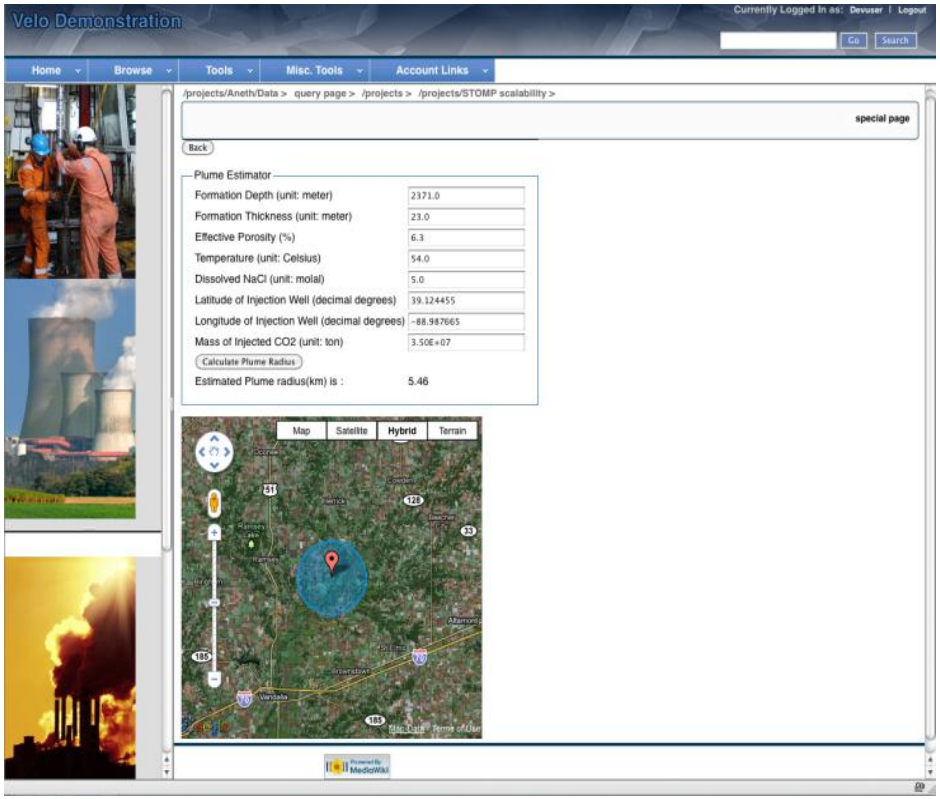
- Mediawiki plugins
- ‘Black box’ tools
- External 3rd party tools





GS3 Examples – Tool Plugins

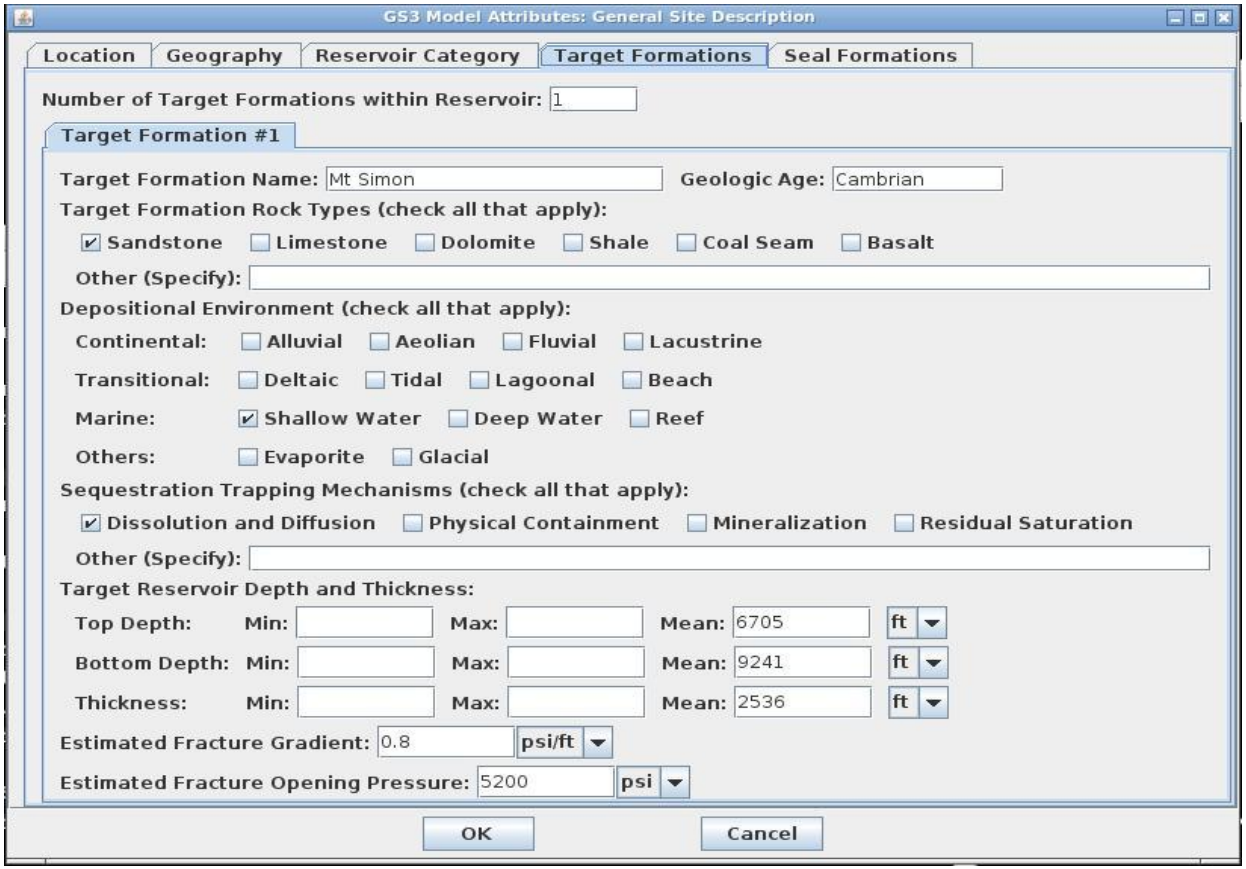




Pacific Northwest
NATIONAL LABORATORY

Proudly Operated by Battelle *Since 1965*

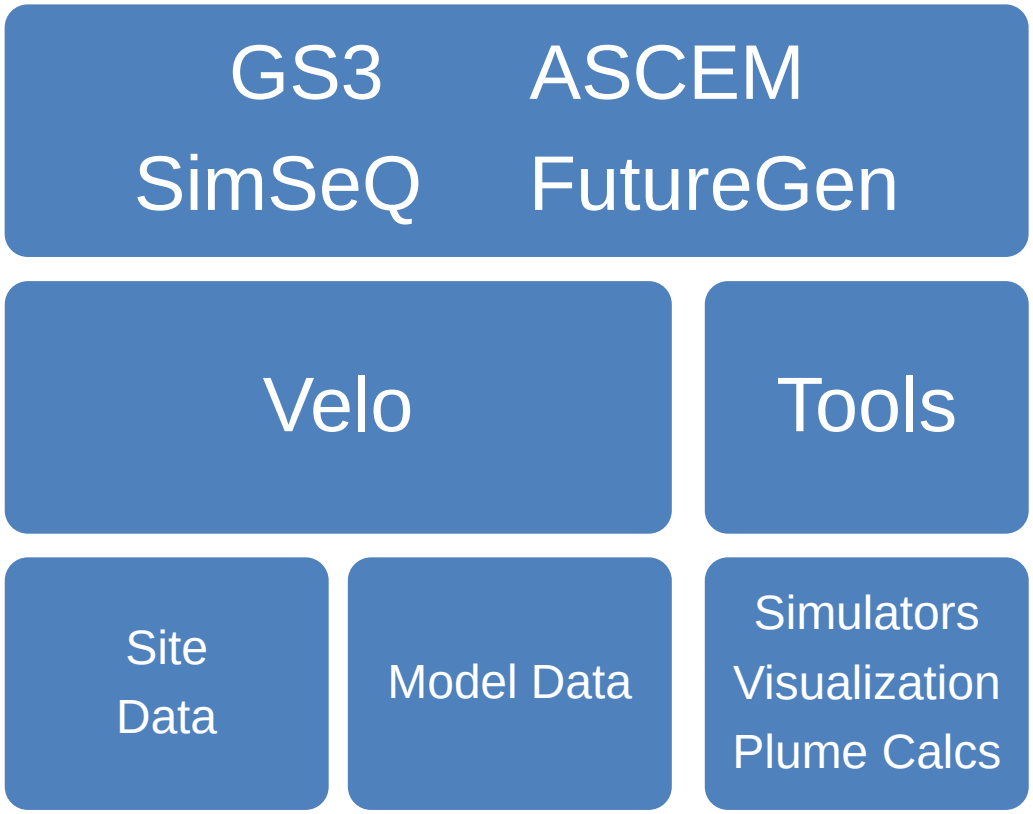
GS3 Examples – Black box Tool Plugins



What Happened?

- Iterative development process
  - Design, build and demo, repeat
- Interest from user community was strong
  - Power of mock-ups and prototypes
- New funding obtained
- Initial sites deployed
- And along the way …

Flexible, Rigorous Scientific Knowledge Management

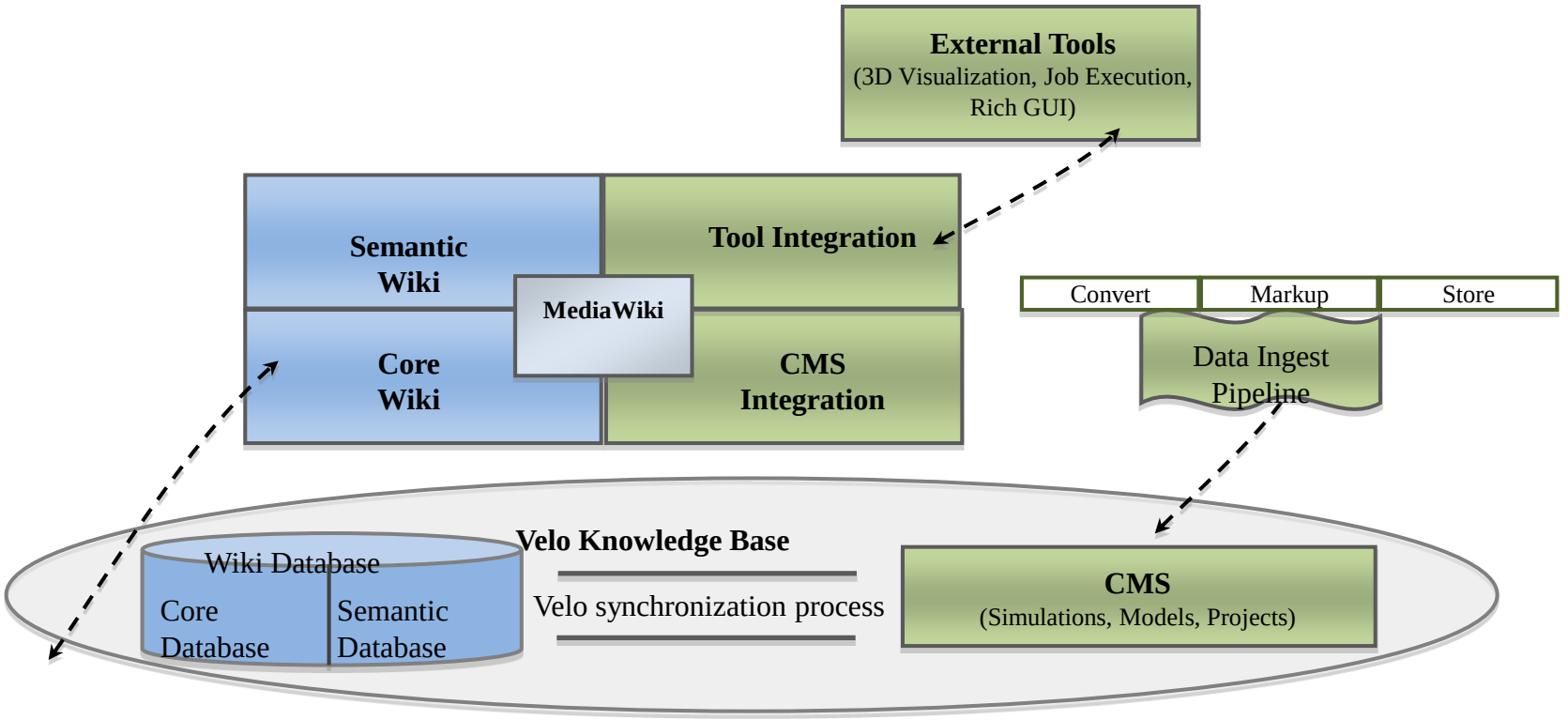


User customizable ‘skins’
Web-based
Extensible

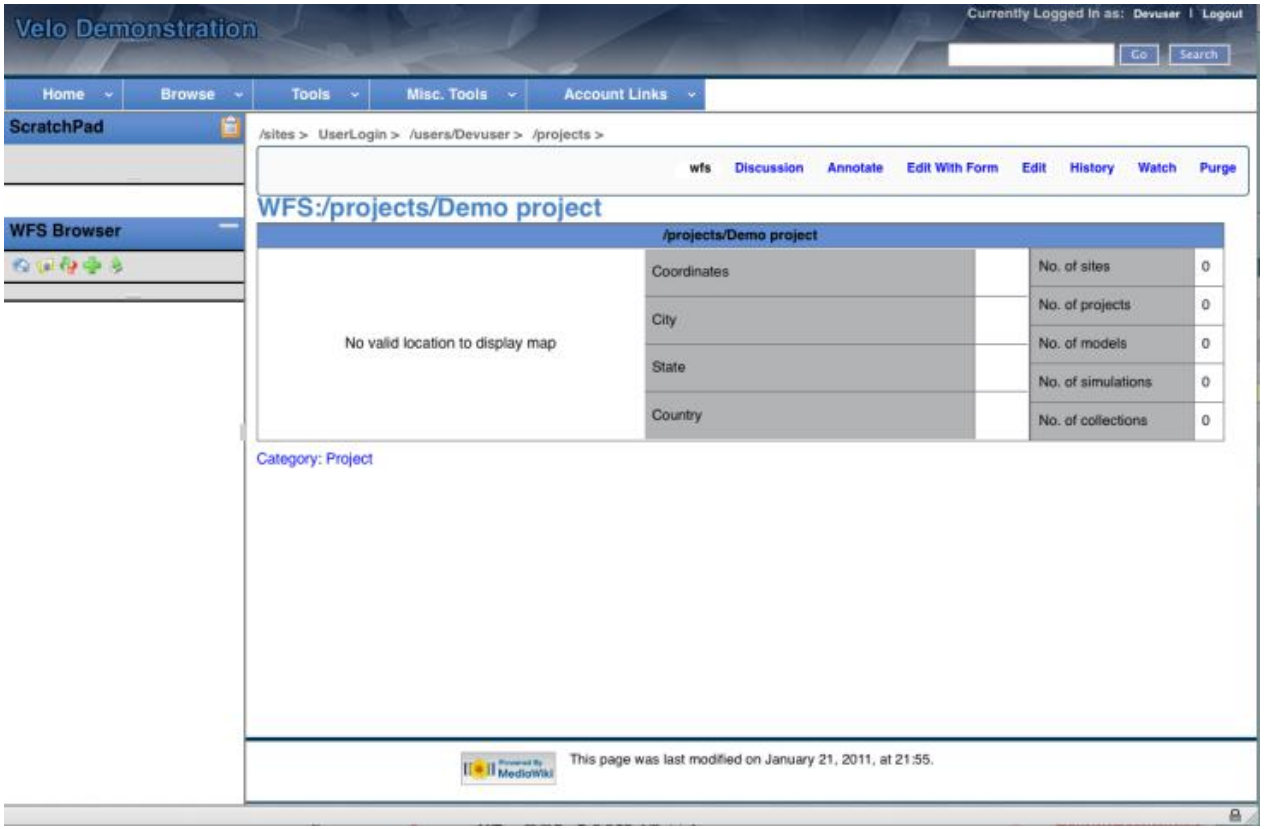
Raw data and metadata storage
Versioning
Provenance
Tool registry
Many deployment options

Extensible data types
Extensible tool repository
Programming interfaces

Velo Architecture



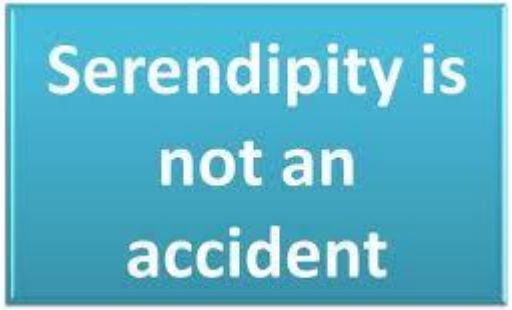
Basic Velo ‘Skin’



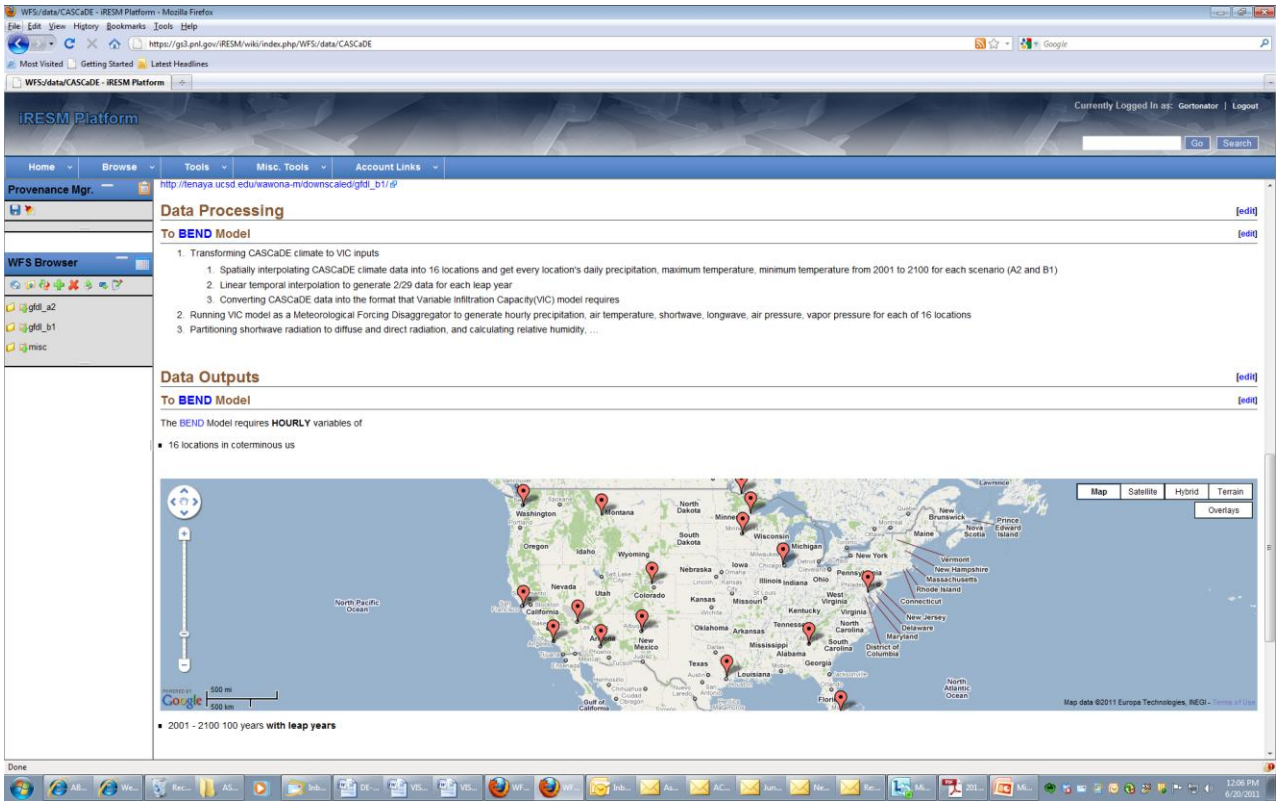
1. Tool Navigation
2. File Browser
3. Wiki Functions
4. Content Viewer

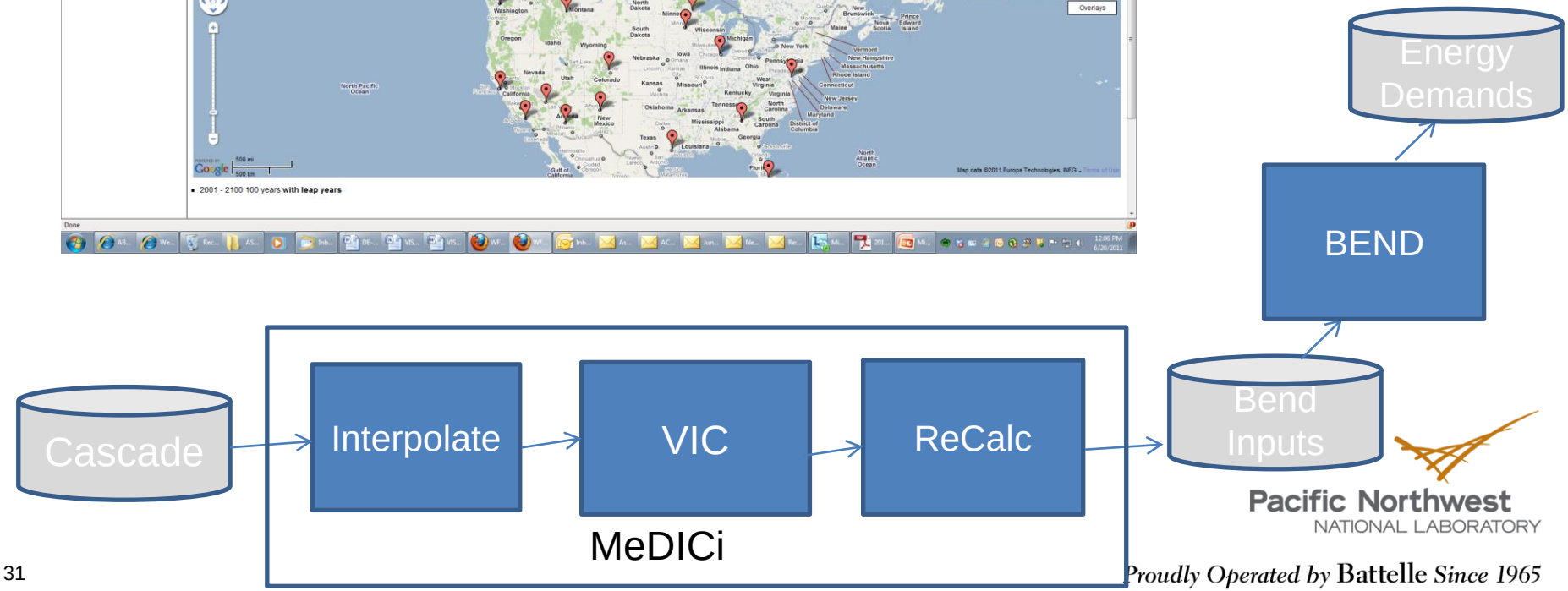
Pacific Northwest
NATIONAL LABORATORY

When requirements collide



Bringing it all Together – iRESM Platform





Pacific Northwest NATIONAL LABORATORY

Proudly Operated by Battelle *Since 1965*

iRESM Feature – Reuse Matrix

Feature	Solution	Notes	Reuse
Collaboration	Velo	Core wiki features support this, includes support for discussions	100%
Sharing data	Velo	Requires integration of MW and Alfresco	100%
Metadata extraction	Velo	Requires creation of data-type specific extraction pipelines	50%
User-driven customization	Velo	Core wiki features support this	100%
Linking to model execution	Velo MeDICi	Need to construct MeDICi pipelines and link to Velo	70%
Model versioning	Velo	Supported	100%
Data set versioning	Velo	Supported	100%
Role-based Security	Velo	Supported	100%

Some reflections

- Science is a complex domain
  - Understanding requirements
  - Diversity of software/data
  - Users who are pushing the boundaries
  - Scientists don't (in general) understand complexity of software systems
  - Architectures, integration, testing
  - Different to implementing a set of equations
- Through deliberate, creative reuse and a strong focus on architecture, we've:
  - Built generically useful technologies at low cost)
  - They work ;)

Questions?